UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/037,894	10/22/2001	Ching-Jer Liang	JCLA7410	6847

23900 7590 09/08/2005

J C PATENTS, INC.
4 VENTURE, SUITE 250
IRVINE, CA 92618

EXAMINER	
THANGAVELU, KANDASAMY	
ART UNIT	**PAPER NUMBER**
2123	

DATE MAILED: 09/08/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/037,894	LIANG, CHING-JER	
	Examiner	Art Unit	
	Kandasamy Thangavelu	2123	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 June 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-11* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-11* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 June 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This communication is in response to the Applicants' Response mailed on June 23, 2005. Claims 1, 3, 4, 6, 7, 9 and 10 were amended. Claims 1-11 of the application are pending. This office action is made final.

Drawings

2. The drawings submitted on June 23, 2005 are accepted.

Specification

3. In the Office action mailed on March 15, 2005 the Examiner indicated numerous errors in the specification and requested necessary corrections. The corrected paragraphs must be provided indicating all changes relative to the immediate prior version of the specification. The text of any added subject matter must be shown by underlining the text. The text of any deleted characters must be shown by strike through.

The applicant corrected most of the errors and provided a supplemental specification. Supplemental specification is not accepted by the PTO. Therefore the supplemental specification submitted by the applicant is **not entered**. The Examiner maintains previous objections to the specification.

However, the applicant may submit a **substitute specification** by stating on a separate sheet that a substitute specification is being submitted incorporating all required corrections under 37 CFR 1.125(b). For the substitute specification to be entered, 37 CFR 1.125 (b) requires that it should be accompanied by **a statement that the substitute specification includes no new matter**. Any substitute specification containing new matter will be denied entry by the Examiner.

37 CFR 1.125 (c) requires a substitute specification filed under 37 CFR 1.125 (b) be submitted in clean form without markings. A marked up copy of the substitute specification showing all changes relative to the immediate prior version of the specification must also be submitted. The text of any added subject matter must be shown by underlining the text. The text of any deleted characters must be shown by strike through. Applicant's attention is directed to MPEP 608.01(q) for further information on submitting substitute specification.

The disclosure is objected to because of the following informalities:

Amended Specification Page 2, Para 0003, Lines 2-3, "so that any problem area can be deal with appropriately" appears to be incorrect and it appears that it should be "so that any problem area can be dealt with appropriately".

Amended Specification Page 2, Para 0003, Lines 14-21 state, "when the instruction counter reaches a first upper value, the microprocessor is triggered into a circuit emulation mode ... and then produced a report ... When the value in the cycle counter reaches a second upper value, the microprocessor is triggered into a circuit

emulation mode...". The terms "a first upper value" and " a second upper value" are **new terms** introduced by the applicant in this amendment to the specification. Such introduction of **new terms and concepts is not allowed** and the applicant is directed to stick with the statements in the original specification. Additionally, whenever a correction is made in the specification, the applicant is required to provide a substitute paragraph for the paragraph being corrected, so the substitute paragraph will be used when the application is printed. Otherwise, the corrections incorporated by the applicant by strikethroughs will not be made by the PTO when the application is printed.

Amended Specification, page 4, Para 0008, Lines state, "when the value inside the instruction counter reaches a first upper value, the microprocessor is triggered into a circuit emulation mode ... when the value inside the cycle counter reaches a second upper value, the microprocessor is triggered into a circuit emulation mode and the values inside...". The terms "a first upper value" and " a second upper value" are **new terms** introduced by the applicant in this amendment to the specification. Such introduction of **new terms and concepts is not allowed** and the applicant is directed to stick with the statements in the original specification. Additionally, whenever a correction is made in the specification, the applicant is required to provide a substitute paragraph for the paragraph being corrected, so the substitute paragraph will be used when the application is printed. Otherwise, the corrections incorporated by the applicant by underscored additions will not be made by the PTO when the application is printed.

Amended specification Page 6, Para 0018, Line 21 refers to "a first upper value and a second upper value respectively". The terms "a first upper value" and " a second upper value" are **new terms** introduced by the applicant in this amendment to the specification. Amended specification Page 7, Para 0018, Line 3 refers to "the first and the second upper values". The term "the first and the second upper values" is a **new term** introduced by the applicant in this amendment to the specification. Such introduction of **new terms and concepts is not allowed** and the applicant is directed to stick with the statements in the original specification.

Appropriate corrections are required.

Claim Objections

4. The following is a quotation of 37 C.F.R § 1.75 (d)(1):

> The claim or claims must conform to the invention as set forth in the remainder of the specification and terms and phrases in the claims must find clear support or antecedent basis in the description so that the meaning of the terms in the claims may be ascertainable by reference to the description.

5. Claims 1-4, and 6-10 are objected to because of the following informalities:

Amended Claims 1-4, and 7-10 have a limitation, "reading out the value inside the instruction counter and the cycle counter". Since there are two counters referred to and each will have a value, it should be "reading out the values inside the instruction counter and the cycle counter".

In Amended Claim 6, "the microprocessor is triggered into the circuit emulation mode after the values within the instruction counter and the cycle counter are read out"

appears to be incorrect and it appears that it should be "the microprocessor is triggered into the circuit emulation mode and the values within the instruction counter and the cycle counter are read out".

Appropriate corrections are required.

Claim Rejections - 35 USC § 112

6. The following is a quotation of the first paragraph of 35 U.S.C. §112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

7. Claims 1- 11 are rejected under 35 U.S.C. 112, first paragraph, as containing subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

7.1. Amended claim 1 states in part, "setting a first upper value in an instruction counter and a second upper value in a cycle counter; ... wherein the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches the first upper value; ... wherein the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches the second upper value".

Amended claim 3 states in part, "wherein the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches a first upper value; ... wherein the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches a second upper value".

Amended claim 6 states in part, "when the instruction counter counts to a first upper value, the microprocessor is triggered into the circuit emulation mode ... when the cycle counter counts to a second upper value".

Amended claim 7 states in part, "setting a first upper value in an instruction counter and a second upper value in a cycle counter; ... when either the instruction counter reaches the first upper value or the cycle counter reaches the second upper value".

The terms "a first upper value" and " a second upper value" are **new terms** not found in the original specification but introduced by the applicant in this amendment to the specification. Such introduction of **new terms and concepts is not allowed** and therefore this amendment is not allowed.

Claims rejected but not specifically addressed are rejected based on their dependency on rejected claims.

8. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

9. Claims 1-11 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 1 recites the limitations " determining the performance of a microprocessor ... initiating the counting by the instruction counter ... initiating the counting by the cycle counter ... reading out the value inside the instruction counter and the cycle counter ". There is insufficient antecedent basis for "the performance", "the counting" and "the value" in the claim.

Claim 2 recites the limitations " reading out the value inside the instruction counter and the cycle counter ". There is insufficient antecedent basis for "the value" in the claim.

Claim 3 recites the limitations " initiating the counting by the instruction counter ... initiating the counting by the cycle counter ... reading out the value inside the instruction counter and the cycle counter ". There is insufficient antecedent basis for "the counting" and "the value" in the claim.

Claim 4 recites the limitations " the start assessment point ... initiating the counting by the instruction counter ... initiating the counting by the cycle counter ... reading out the value inside the instruction counter and the cycle counter ". There is insufficient antecedent basis for "the start assessment point", "the counting" and "the value" in the claim.

Claim 5 recites the limitations " the evaluation of microprocessor ... dividing the value inside cycle counter by the value inside the instruction counter ". There is insufficient antecedent basis for "the evaluation of microprocessor", "the value" and "the value" in the claim.

Claim 6 recites the limitations " determining the performance of a microprocessor ... the values within the instruction counter and the cycle counter ... the value inside the cycle counter by the value inside the instruction counter ". There is insufficient antecedent basis for "the performance", "the values" and "the value" in the claim.

Claim 7 recites the limitations " determining the performance of a microprocessor ... initiating the counting by the instruction counter ... reading out the value inside the instruction counter and the cycle counter ". There is insufficient antecedent basis for "the performance", "the counting" and "the value" in the claim.

Claim 8 recites the limitations " reading out the value inside the instruction counter and the cycle counter ". There is insufficient antecedent basis for "the value" in the claim.

Claim 9 recites the limitations " reading out the value inside the instruction counter and the cycle counter ". There is insufficient antecedent basis for "the value" in the claim.

Claim 10 recites the limitations " the start assessment point ... the ending assessment point ... reading out the value inside the instruction counter and the cycle counter ". There is

insufficient antecedent basis for "the start assessment point", "the ending assessment point" and "the value" in the claim.

Claim 11 recites the limitations " dividing the value inside cycle counter by the value inside the instruction counter ". There is insufficient antecedent basis for "the value" and "the value" in the claim.

Claims rejected but not specifically addressed are rejected based on their dependency on rejected claims.

10. Claims 1-5 are rejected under 35 U.S.C. § 112, second paragraph, as being incomplete for omitting essential steps, such omission amounting to a gap between the steps. See MPEP § 2172.01. The omitted steps are:

Claim1 states in part, "initiating the counting by the instruction counter such that the instruction counter increments by one when an instruction is executed, wherein the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches the first upper value". It does not state what is done once the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches the first upper value. It appears that a step is missing here.

Claim1 states in part, "initiating the counting by the cycle counter such that the cycle counter increments by one when a timing pulse traverses a cycle, wherein the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches the second upper value". It does not state what is done once the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches the second upper value. It appears that a step is missing here.

Claim3 states in part, "initiating the counting by the instruction counter such that the instruction counter increments by one when an instruction is executed, wherein the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches a first upper value". It does not state what is done once the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches a first upper value. It appears that a step is missing here.

Claim3 states in part, "initiating the counting by the cycle counter such that the cycle counter increments by one when a timing pulse traverses a cycle, wherein the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches a second upper value". It does not state what is done once the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches a second upper value. It appears that a step is missing here.

Claims rejected but not specifically addressed are rejected based on their dependency on rejected claims.

11. Claim 6 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 6 states in part, "while a cycle counter for counting up by one whenever one cycle of timing pulse is traversed, and when the cycle counter counts to a second upper value, wherein the instruction counter and the cycle counter start to count concurrently, and the microprocessor is triggered into the circuit emulation mode after the values within the instruction counter and the cycle counter are read out'. The above statement has several clauses connected incoherently making it difficult to figure out the intent of the applicant. Therefore, the above statement is indefinite.

Claim Rejections - 35 USC § 103

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.

13. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

14. Claims 1, 6 and 7 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Berc et al.** (U.S. Patent 6,112,317) in view of **Killian et al.** (U.S. Patent 6,477,683).

14.1 **Berc et al.** teaches processor performance counter for sampling the execution frequency of individual instructions. Specifically as per claim 1, **Berc et al.** teaches a method of determining the performance of a microprocessor, wherein the performance of a program having a plurality of instructions is assessed (Abstract, L3-5; CL1, L43-50; CL1, L54-56); comprising the steps of:

setting a first upper value in an instruction counter and a second upper value in a cycle counter (Abstract, L8-10; CL2, L8-11);

resetting an instruction counter and a cycle counter to zero (CL1, L43-50; CL3, L14-19; CL7, L11-14);

initiating the counting by the instruction counter such that the instruction counter increments by one when an instruction is executed (Abstract, L5-8; CL1, L43-50; CL3, L14-19; CL7, L31-33);

initiating the counting by the cycle counter such that the cycle counter increments by one when a timing pulse traverses a cycle (CL7, L11-14; CL1, L43-50);

the instruction counter and the cycle counter start to count concurrently (CL3, L21-22; CL4, L50-53; CL5, L4-8);

reading out the value inside the instruction counter and the cycle counter(CL3, L21-22); and

evaluating microprocessor performance (CL4, L51-53; CL5, L4-8).

Berc et al. does not expressly teach microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode. **Killian et al.** teaches microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

Berc et al. does not expressly teach triggering the microprocessor into the normal operating mode and executing the program. **Killian et al.** teaches triggering the microprocessor

into the normal operating mode and executing the program (CL13, L19-23), because that allows control of execution of the processor using the debugger and the capability of the on-chip debug mode (CL13, L30-35). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the normal operating mode and executing the program. The artisan would have been motivated because that would allow control of execution of the processor using the debugger and the capability of the on-chip debug mode.

Berc et al. teaches that the microprocessor is triggered to generate an interrupt to an interrupt handler when the instruction counter reaches the first upper value; and the microprocessor is triggered to generate an interrupt to an interrupt handler when the cycle counter reaches the second upper value (Abstract, L8-10; CL3, L19-20). **Berc et al.** does not expressly teach that the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches the first upper value; and the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches the second upper value. **Killian et al.** teaches that the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches the first upper value; and the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches the second upper value (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included the microprocessor being triggered into the circuit emulation mode when the instruction counter

reaches the first upper value; and the microprocessor being triggered into the circuit emulation mode when the cycle counter reaches the second upper value. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode when the program is executed to an assessment point. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the program is executed to an assessment point (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode when the program is executed to an assessment point. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

14.2 As per claim 6, **Berc et al.** teaches a device for determining the performance of a microprocessor execution (Abstract, L3-5; CL1, L43-50; CL1, L54-56); comprising:

an instruction counter for counting up by one whenever an instruction is executed (Abstract, L5-8; CL1, L43-50; CL3, L14-19; CL7, L31-33);

a cycle counter for counting up by one whenever one cycle of timing pulse is traversed (CL7, L11-14; CL1, L43-50);

the instruction counter and the cycle counter start to count concurrently (CL3, L21-22; CL4, L50-53; CL5, L4-8);

the values within the instruction counter and the cycle counter are read out (CL3, L21-22); and

microprocessor performance is evaluated by dividing the value inside the cycle counter by the value inside the instruction counter (CL4, L51-53; CL5, L4-8).

Berc et al. does not expressly teach a microprocessor capable of operating in a circuit emulation mode and a normal operating mode. **Killian et al.** teaches a microprocessor capable of operating in a circuit emulation mode and a normal operating mode (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the device of **Berc et al.** with the device of **Killian et al.** that included a microprocessor capable of operating in a circuit emulation mode and a normal operating mode. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

Berc et al. teaches that the microprocessor is triggered to generate an interrupt to an interrupt handler when the instruction counter reaches a first upper value; and the microprocessor is triggered to generate an interrupt to an interrupt handler when the cycle counter reaches a second upper value (Abstract, L8-10; CL3, L19-20). **Berc et al.** does not expressly teach that

when the instruction counter counts to a first upper value, the microprocessor is triggered into the circuit emulation mode; and when the cycle counter counts to a second upper value, the microprocessor is triggered into the circuit emulation mode. **Killian et al.** teaches that when the instruction counter counts to a first upper value, the microprocessor is triggered into the circuit emulation mode; and when the cycle counter counts to a second upper value, the microprocessor is triggered into the circuit emulation mode (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the device of **Berc et al.** with the device of **Killian et al.** that included when the instruction counter counts to a first upper value, the microprocessor is triggered into the circuit emulation mode; and when the cycle counter counts to a second upper value, the microprocessor is triggered into the circuit emulation mode. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

14.3 As per claim 7, **Berc et al.** teaches a method of determining the performance of a microprocessor, wherein the performance of a program having a plurality of instructions is assessed (Abstract, L3-5; CL1, L43-50; CL1, L54-56); comprising the steps of:

setting a first upper value in an instruction counter and a second upper value in a cycle counter (Abstract, L8-10; CL2, L8-11);

resetting an instruction counter and a cycle counter to zero (CL1, L43-50; CL3, L14-19; CL7, L11-14);

initiating the counting by either the instruction counter such that the instruction counter increments by one when an instruction is executed (Abstract, L5-8; CL1, L43-50; CL3, L14-19; CL7, L31-33); or by the cycle counter such that the cycle counter increments by one when a timing pulse traverses a cycle (CL7, L11-14; CL1, L43-50);

the instruction counter and the cycle counter start to count concurrently (CL3, L21-22; CL4, L50-53; CL5, L4-8);

reading the value inside the instruction counter and the cycle counter to evaluate execution performance (CL3, L21-22);

reading out the value inside the instruction counter and the cycle counter(CL3, L21-22); and

evaluating microprocessor performance (CL4, L51-53; CL5, L4-8).

Berc et al. does not expressly teach microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode. **Killian et al.** teaches microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included

microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

Berc et al. does not expressly teach triggering the microprocessor into the normal operating mode and executing the program. **Killian et al.** teaches triggering the microprocessor into the normal operating mode and executing the program (CL13, L19-23), because that allows control of execution of the processor using the debugger and the capability of the on-chip debug mode (CL13, L30-35). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the normal operating mode and executing the program. The artisan would have been motivated because that would allow control of execution of the processor using the debugger and the capability of the on-chip debug mode.

Berc et al. teaches that the microprocessor is triggered to generate an interrupt to an interrupt handler when the instruction counter reaches the first upper value; and the microprocessor is triggered to generate an interrupt to an interrupt handler when the cycle counter reaches the second upper value (Abstract, L8-10; CL3, L19-20). **Berc et al.** does not expressly teach that when either the instruction counter reaches the first upper value or the cycle counter reaches the second upper value, the microprocessor is triggered into the circuit emulation mode. **Killian et al.** teaches that when either the instruction counter reaches the first upper value or the cycle counter reaches the second upper value, the microprocessor is triggered into the circuit emulation mode (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip

debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included when either the instruction counter reached the first upper value or the cycle counter reached the second upper value, the microprocessor was triggered into the circuit emulation mode. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

Berc et al. does not expressly teach setting up an assessment point into a breaking point register. **Killian et al.** teaches setting up an assessment point into a breaking point register (CL31, L36-38), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included setting up an assessment point into a breaking point register. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode when the program is executed to the assessment point. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the program is executed to the assessment point (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the

states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode when the program is executed to the assessment point. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

15. Claims 2-5 and 8 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Berc et al.** (U.S. Patent 6,112,317) in view of **Killian et al.** (U.S. Patent 6,477,683), and further in view of **Doing et al.** (U.S. Patent 6,018,759).

15.1 As per claim 2, **Berc et al.** and **Killian et al.** teach the method of claim 1. **Berc et al.** does not expressly teach triggering the microprocessor into the circuit emulation mode on complete execution of the program; reading out the value inside the instruction counter and the cycle counter; and evaluating microprocessor performance. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the program is executed to a definite point (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode when the program is executed to a definite point. The artisan would have been motivated because that would allow

sampling the counters of the processor at those points; and comparing the states at the breakpoint.

Doing et al. teaches the microprocessor identifying completion of execution of the program; reading out the value inside the instruction counter and the cycle counter; and evaluating microprocessor performance (CL22, L21-28), because that allows determining the relative performance of target program during the execution (CL22, L21-22). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Doing et al.** that included the microprocessor identifying completion of execution of the program; reading out the value inside the instruction counter and the cycle counter; and evaluating microprocessor performance. The artisan would have been motivated because that would allow determining the relative performance of target program during the execution.

15.2 As per claim 3, **Berc et al., Killian et al.** and **Doing et al.** teach the method of claim 2. **Berc et al.** teaches initiating the counting by the instruction counter such that the instruction counter increments by one when an instruction is executed (Abstract, L5-8; CL1, L43-50; CL3, L14-19; CL7, L31-33);

initiating the counting by the cycle counter such that the cycle counter increments by one when a timing pulse traverses a cycle (CL7, L11-14; CL1, L43-50);

the instruction counter and the cycle counter start to count concurrently (CL3, L21-22; CL4, L50-53; CL5, L4-8);

reading out the value inside the instruction counter and the cycle counter (CL3, L21-22); and

evaluating microprocessor performance (CL4, L51-53; CL5, L4-8).

Berc et al. does not expressly teach setting up an assessment point into a breaking point register where performance measurement is required when the microprocessor is in the circuit emulation mode. **Killian et al.** teaches setting up an assessment point into a breaking point register where performance measurement is required when the microprocessor is in the circuit emulation mode (CL31, L36-38), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included setting up an assessment point into a breaking point register where performance measurement is required when the microprocessor is in the circuit emulation mode. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

Berc et al. does not expressly teach triggering the microprocessor into the normal operating mode and executing the program. **Killian et al.** teaches triggering the microprocessor into the normal operating mode and executing the program (CL13, L19-23), because that allows control of execution of the processor using the debugger and the capability of the on-chip debug mode (CL13, L30-35). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.**

that included triggering the microprocessor into the normal operating mode and executing the program. The artisan would have been motivated because that would allow control of execution of the processor using the debugger and the capability of the on-chip debug mode.

Berc et al. teaches that the microprocessor is triggered to generate an interrupt to an interrupt handler when the instruction counter reaches a first upper value; and the microprocessor is triggered to generate an interrupt to an interrupt handler when the cycle counter reaches a second upper value (Abstract, L8-10; CL3, L19-20). **Berc et al.** does not expressly teach that the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches a first upper value; and the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches a second upper value. **Killian et al.** teaches that the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches a first upper value; and the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches a second upper value (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included the microprocessor being triggered into the circuit emulation mode when the instruction counter reaches a first upper value; and the microprocessor being triggered into the circuit emulation mode when the cycle counter reaches a second upper value. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode when the assessment point is encountered during instruction execution. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the assessment point is encountered during instruction execution (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode when the assessment point is encountered during instruction execution. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

15.3 As per claim 4, **Berc et al., Killian et al.** and **Doing et al.** teach the method of claim 3. **Berc et al.** teaches reading out the value inside the instruction counter and the cycle counter (CL3, L21-22);

resetting an instruction counter and a cycle counter to zero (CL1, L43-50; CL3, L14-19; CL7, L11-14);

initiating the counting by the instruction counter such that the instruction counter increments by one when an instruction is executed (Abstract, L5-8; CL1, L43-50; CL3, L14-19; CL7, L31-33);

initiating the counting by the cycle counter such that the cycle counter increments by one when a timing pulse traverses a cycle (CL7, L11-14; CL1, L43-50); and

the instruction counter and the cycle counter start to count concurrently (CL3, L21-22; CL4, L50-53; CL5, L4-8);

evaluating microprocessor performance (CL4, L51-53; CL5, L4-8).

Berc et al. does not expressly teach setting up an assessment point into the breaking point register where performance measurement is required. **Killian et al.** teaches setting up an assessment point into the breaking point register where performance measurement is required (CL31, L36-38), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included setting up an assessment point into the breaking point register where performance measurement is required. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode when the start assessment point is encountered during instruction execution. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the start assessment point is encountered during instruction execution (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It

would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode when the start assessment point is encountered during instruction execution. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

Berc et al. does not expressly teach triggering the microprocessor into the normal operating mode and executing the program. **Killian et al.** teaches triggering the microprocessor into the normal operating mode and executing the program (CL13, L19-23), because that allows control of execution of the processor using the debugger and the capability of the on-chip debug mode (CL13, L30-35). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.**

that included triggering the microprocessor into the normal operating mode and executing the program. The artisan would have been motivated because that would allow control of execution of the processor using the debugger and the capability of the on-chip debug mode.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode when the assessment point is encountered. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the assessment point is encountered (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode when the assessment point is encountered. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

Per claim 5: **Berc et al.** teaches dividing the value inside cycle counter by the value inside the instruction counter (CL4, L51-53; CL5, L4-8).

15.4 As per Claim 8, it is rejected based on the same reasoning as Claim 2, supra. Claim 8 is a method claim reciting the same limitations as Claim 2, as taught throughout by **Berc et al., Killian et al.** and **Doing et al.**

15.5 As per claim 9, **Berc et al., Killian et al.** and **Doing et al.** teach the method of claim 8. **Berc et al.** teaches reading out the value inside the instruction counter and the cycle counter (CL3, L21-22); and

evaluating microprocessor performance (CL4, L51-53; CL5, L4-8).

Berc et al. does not expressly teach setting up an assessment point into a breaking point register where performance measurement is required when the microprocessor is in the circuit emulation mode. **Killian et al.** teaches setting up an assessment point into a breaking point register where performance measurement is required when the microprocessor is in the circuit emulation mode (CL31, L36-38), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included setting up an assessment point into a breaking point register where performance measurement is required when the microprocessor is in the circuit emulation mode. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode when the assessment point is encountered during instruction execution. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the assessment point is encountered during instruction execution (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31,

L63-66); and comparing the states at the breakpoint (CL31, L46-49). It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included triggering the microprocessor into the circuit emulation mode when the assessment point is encountered during instruction execution. The artisan would have been motivated because that would allow sampling the counters of the processor at those points; and comparing the states at the breakpoint.

15.6 As per Claims 10-11, these are rejected based on the same reasoning as Claims 4-5, supra. Claims 10-11 are method claims reciting the same limitations as Claims 4-5, as taught throughout by **Berc et al., Killian et al.,** and **Doing et al.**

Response to Arguments

16. Applicants' arguments filed on June 23, 2005 have been fully considered. The arguments with respect to 103 (a) rejections are not persuasive, in view of new rejections made against the amended claims.

16.1 As per the applicants' argument that "Berc discloses a processor performance counter for determining a performance of each instruction; Killian provides a system which can optimize the hardware implementation and software tools for various performance criteria; since objectives of Berc are different than those of Killian, there is no motive for any one skilled in the art to

combine Berc and Killian to arrive at the subject matter of the amended claim 1, 6 and 7", the examiner respectfully disagrees.

Berc et al. teaches a method of determining the performance of a microprocessor, wherein the performance of a program having a plurality of instructions is assessed (Abstract, L3-5; CL1, L43-50; CL1, L54-56); comprising the steps of:

setting a first upper value in an instruction counter and a second upper value in a cycle counter (Abstract, L8-10; CL2, L8-11); resetting an instruction counter and a cycle counter to zero (CL1, L43-50; CL3, L14-19; CL7, L11-14); initiating the counting by the instruction counter such that the instruction counter increments by one when an instruction is executed (Abstract, L5-8; CL1, L43-50; CL3, L14-19; CL7, L31-33); initiating the counting by the cycle counter such that the cycle counter increments by one when a timing pulse traverses a cycle (CL7, L11-14; CL1, L43-50); the instruction counter and the cycle counter start to count concurrently (CL3, L21-22; CL4, L50-53; CL5, L4-8); reading out the value inside the instruction counter and the cycle counter(CL3, L21-22); and evaluating microprocessor performance (CL4, L51-53; CL5, L4-8).

Berc et al. does not expressly teach microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode. **Killian et al.** teaches microprocessor having a circuit emulation mode and a normal operating mode; and triggering the microprocessor into the circuit emulation mode (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state

of the microprocessor including all program visible registers or memory locations (CL13, L15-19).

Berc et al. does not expressly teach triggering the microprocessor into the normal operating mode and executing the program. **Killian et al.** teaches triggering the microprocessor into the normal operating mode and executing the program (CL13, L19-23), because that allows control of execution of the processor using the debugger and the capability of the on-chip debug mode (CL13, L30-35).

Berc et al. teaches that the microprocessor is triggered to generate an interrupt to an interrupt handler when the instruction counter reaches the first upper value; and the microprocessor is triggered to generate an interrupt to an interrupt handler when the cycle counter reaches the second upper value (Abstract, L8-10; CL3, L19-20). **Berc et al.** does not expressly teach that the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches the first upper value; and the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches the second upper value. **Killian et al.** teaches that the microprocessor is triggered into the circuit emulation mode when the instruction counter reaches the first upper value; and the microprocessor is triggered into the circuit emulation mode when the cycle counter reaches the second upper value (CL4, L34-39; CL13, L14-18), because the emulation mode (on-chip debug mode) allows accessing the internal state of the microprocessor including all program visible registers or memory locations (CL13, L15-19).

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode when the program is executed to an assessment point. **Killian et al.** teaches

triggering the microprocessor into the circuit emulation mode when the program is executed to an assessment point (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49).

It would have been obvious to one of ordinary skill in the art at the time of Applicants' invention to modify the method of **Berc et al.** with the method of **Killian et al.** that included the microprocessor being triggered into the circuit emulation mode when the instruction counter reaches the first upper value; and the microprocessor being triggered into the circuit emulation mode when the cycle counter reaches the second upper value. The artisan would have been motivated because the emulation mode (on-chip debug mode) would allow accessing the internal state of the microprocessor including all program visible registers or memory locations.

16.2 As per the applicants' argument that "in Killian, a debug mode functions to access the internal, software-visible state of the processor, which is distinct from "a circuit emulation mode" as claimed in the amended claims 1, 6 and 7; the combination of Berc and Killian still fail to teach, suggest or disclose setting a first upper value in an instruction counter and a second value in a cycle counter when the microprocessor is in the circuit emulation mode, and the microprocessor is triggered into the circuit emulation mode when either the instruction counter reaches the first upper value or the cycle counter reaches the second upper value as claimed in the amended claims 1, 6 and 7; the combination of Berc and Killian still fail to teach, suggest or disclose triggering the microprocessor into the circuit emulation mode when the program is executed to an assessment point as claimed in the amended claims 1, 6 and 7; the combination of

Berc and Killian still fail to teach, suggest or disclose the microprocessor jumps from the normal operating mode into the circuit emulation mode in case of the instructor counter's counting to its preset first upper value, the cycle counter's counting to its preset second upper value or the breaking register's monitoring the program is executed to a preset assessment point, as claimed in the amended claims 1, 6 and 7; the combination of Berc and Killian still fail to teach, suggest or disclose setting an assessment point into a breaking register when the microprocessor is in the circuit emulation mode as claimed in the amended claim 7", the examiner directs applicant's attention to Paragraph 16.1 above.

In addition, the terms "first upper value" and "second upper value" are new terms introduced by the applicant in the amendment to the application made on June 23, 2005; these terms are not in the original specification and therefore are not allowed as indicated in Paragraphs 3 and 7.1.

Berc et al. does not expressly teach setting up an assessment point into a breaking point register. **Killian et al.** teaches setting up an assessment point into a breaking point register (CL31, L36-38), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49).

16.3 As per the applicants' argument that "Doing functions to provide an improved data processing system and method for multithreaded processor embodied in the hardware of the processor, and fails to mention the instruction counter's preset first upper value and the cycle counter's preset second upper value as disclosed in the amended claims 1,6and 7; therefore, the

combination of Berc, Killian and Doing still fail to teach , suggest or disclose triggering the microprocessor into the circuit emulation mode on complete execution of the program as claimed in the claims 2 and 8; ... the combination of Berc and Killian still fail to teach, suggest or disclose the microprocessor jumps from the normal operating mode into the circuit emulation mode in case of the instructor counter's counting to its preset first upper value, the cycle counter's counting to its preset second upper value or the breaking register's monitoring the program is executed to a preset assessment point, as claimed in the amended claims 1, 6 and 7; even this combination incorporating Doing, it still fails to make the preceding features of the amended claims 1, 6 and 7", the examiner directs applicant's attention to Paragraph 16.2 above regarding the first upper value and the second upper value being new matter.

Berc et al. does not expressly teach triggering the microprocessor into the circuit emulation mode on complete execution of the program; reading out the value inside the instruction counter and the cycle counter; and evaluating microprocessor performance. **Killian et al.** teaches triggering the microprocessor into the circuit emulation mode when the program is executed to a definite point (CL11, L66-67; CL29, L47-50; CL31, L36-38 and L46-49), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49).

Doing et al. teaches the microprocessor identifying completion of execution of the program; reading out the value inside the instruction counter and the cycle counter; and evaluating microprocessor performance (CL22, L21-28), because that allows determining the relative performance of target program during the execution (CL22, L21-22).

16.4 As per the applicants' argument that "In re claims 3, 4, 9 and 10, the subject matters of theses claims are to setting up an assessment point into a breaking register where performance measurement is required when the microprocessor is in the circuit emulation mode" as claimed in the amended claims 3, 4, 9 and 10. ... Roth only extracting state information from the processor, instead of stating when and how to set up an assessment point", the examiner had shown that **Killian et al.** also used the breakpoint register to set the assessment point.

Berc et al. does not expressly teach setting up an assessment point into a breaking point register. **Killian et al.** teaches setting up an assessment point into a breaking point register (CL31, L36-38), because that allows sampling the counters of the processor at those points (CL31, L63-66); and comparing the states at the breakpoint (CL31, L46-49).

Conclusion

ACTION IS FINAL

17. Applicant's amendments necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO

MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

18. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dr. Kandasamy Thangavelu whose telephone number is 571-272-3717. The examiner can normally be reached on Monday through Friday from 8:00 AM to 5:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Leo Picard, can be reached on 571-272-3749. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

K. Thangavelu
Art Unit 2123
August 31, 2005

Paul L. Rodriguez 9/6/05
Primary Examiner
Art Unit 2125